FACULTY OF INDUSTRIAL SCIENCES & TECHNOLOGY
FINAL EXAMINATION

COURSE	**:**	**NUMERICAL METHODS**
COURSE CODE	**:**	**BUM2313**
LECTURER	**:**	**NURFATIHAH BINTI MOHD HANAFI** **NORHAYATI BINTI ROSLI**
DATE	**:**	**8 JUNE 2016**
DURATION	**:**	**3 HOURS**
SESSION/SEMESTER	**:**	**SESSION 2015/2016 SEMESTER II**
PROGRAMME CODE	**:**	**BAA/BEE/BEP/BFF/BMA/BMM**

INSTRUCTIONS TO CANDIDATE

1. This question paper consists of **SIX (6)** questions. Answer **ALL** questions.
2. Use **FOUR (4) decimal places** in all calculations.
3. All answers to a new question should start on a new page.
4. All the calculations and assumptions must be clearly stated.

EXAMINATION REQUIREMENTS

1. Scientific Calculator
2. **APPENDIX**

DO NOT TURN THIS PAGE UNTIL YOU ARE TOLD TO DO SO

This examination paper consists of **THIRTEEN (13)** printed pages including front page.

QUESTION 1

Consider the following system of linear equations

$$
\begin{aligned}
2x_1 + 8x_2 + 3x_3 + x_4 &= -2 \\
2x_2 - x_3 + 4x_4 &= 4 \\
7x_1 - 2x_2 + x_3 + 2x_4 &= 3 \\
-x_1 + 5x_3 + 2x_4 &= 5
\end{aligned}
$$

(i) Transform the above system of linear equations into the matrix form $A\mathbf{x} = \mathbf{b}$.

(1 Mark)

(ii) Derive the Gauss-Seidel formula based on the system of linear equations in (i).

(6 Marks)

(iii) Find the solution of the system of linear equations by using Gauss-Seidel method with initial vector $x_i^0 = \begin{pmatrix} 0 & -1 & 1 & 0 \end{pmatrix}^T$. Calculate until two iterations.

(7 Marks)

[14 Marks, CO2/PO2]

QUESTION 2

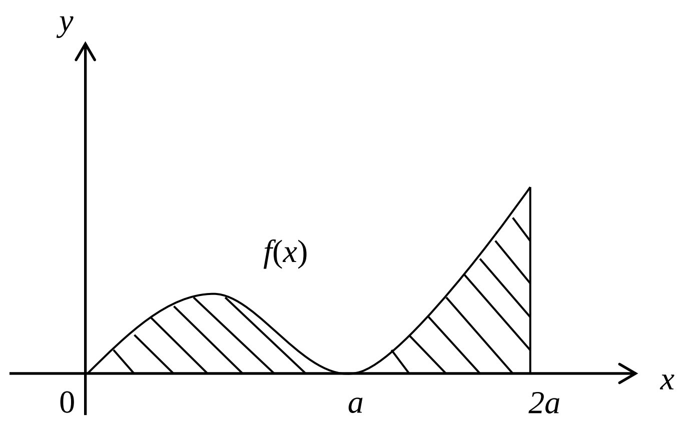

Figure 1

Figure 1 shows the region bounded by the curve of the function

$$f(x) = x(x-2)^2$$

and the line $y = 0$, for $0 \le x \le 2a.$ By using Simpson's rule, estimate the area of shaded region

$$A = \int_0^{2a} f(x)\,dx$$

with $n = 9.$

[17 Marks, CO2/PO2]

QUESTION 3

A mass balance for a chemical in a completely mixed reactor can be written as

$$V\frac{dc}{dt} + kVc^2 = F - Qc$$

where V is volume of a reactor, c refers to the concentration of a chemical in a reactor, F is a feed rate, Q is a flow rate and k represents a coefficient of the second-order reaction rate. Suppose $V = 12\ \text{m}^3$, $F = 175\ \text{g/min}$, $Q = 1\ \text{m}^3/\text{min}$ and $k = 0.15$. If the intial condition is given by $c(0) = 0$, compute the concentration of the chemical in a reactor at $t = 0.5$ minutes, by using fourth order of Runge-Kutta method. Use a time step, $h = 0.5$.

[12 Marks, CO2/PO2]

QUESTION 4

A Lotka-Volterra model is given by

$$\frac{dQ}{dt} = \alpha Q - \beta QP$$
$$\frac{dP}{dt} = \lambda QP - \gamma P$$

where $Q(t)$ and $P(t)$ denote the population of prey and predator at time, t (in month), respectively. Suppose that:

α : the increase rate of prey in the absence of predator.

γ : the death rate of preadator.

β : the death rate of prey due to being eaten by predator.

λ : skill of the predator in catching the prey.

If the initial populations of prey and predator are $Q(0)=1000$ and $P(0)=18$, respectively, with $\alpha=0.2$, $\beta=0.01$, $\lambda=0.02$ and $\gamma=0.4$, find the numbers of prey and predator for $0\le t\le 1$ month. Use the Euler's method with a step size of 0.5.

[12 Marks, CO2/PO2]

QUESTION 5

Figure 2 represents an automobile of mass, m that is supported by springs and shock absorbers. Shock absorbers offer resistance to the motion that is proportional to the vertical speed (up and down motion).

Figure 2

According to Hooke's law, the resistance of the spring is proportional to the spring constant, k and the distance from the equilibrium position, x. Therefore, the spring force, F_s is formulated as

$$F_s=-kx$$

where the negative sign indicates that the restoring force acts to return the car towards the position of equilibrium (negative x direction).

The damping force of the shock absorbers, F_D is given by

$$F_D = -c\frac{dx}{dt}$$

where c is a damping coefficient and $\frac{dx}{dt}$ is the vertical velocity. The negative sign indicates that the damping force acts in the opposite direction against velocity. The motion for the system is given by

$$\text{Mass} \times \text{Acceleration} = \text{Spring force} + \text{Damping force.}$$

Mathematically, it can be written as

$$m\frac{d^2x}{dt^2} = F_s + F_D$$

or

$$m\frac{d^2x}{dt^2} + c\frac{dx}{dt} + kx = 0 \qquad (1)$$

with boundary conditions, $x(0) = 0$ and $x(1.0) = 1.0$. Find the solutions of equation (1) by using Shooting method where the mass of the car, $m = 1.2 \times 10^6$ kg, the spring constant, $k = 1.397 \times 10^9$ kg/s^2 and it has a shock system with a damping coefficient of $c = 1 \times 10^7$. Use a step size, $h = 0.5$ with the first guess, $z(0) = 0.7$ and second guess $z(0) = -0.7$.

[21 Marks, CO2/PO2]

QUESTION 6

Let consider the problem given in **Question 5**, but for the case where the car is subjected to an external force given by

$$P = P_m \sin(\varpi t)$$

where P_m is a forcing coefficient and ϖ is a forcing frequency. The motion for the system is given by

Mass $\times$ Acceleration = Spring force + Damping force + External force.

Thus, the governing differential equation for this case can be written as

$$m\frac{d^2x}{dt^2} = -c\frac{dx}{dt} - kx + P_m \sin(\varpi t).$$

Suppose the mass of the car, $m = 1.5\times10^6$ kg, the spring constant, $k = 2.312\times10^8$ kg/s^2, the forcing coefficient, $P_m = 2.5\times10^7$, the forcing frequency, $\varpi = 0.5$, the damping coefficient of $c = 1\times10^7$ and the boundary conditions, $x(0) = 0$ and $x(2.0) = 3.0$.

(i) Perform discretization process for the linear second order differential equations by dividing the time interval into five equal subintervals.

(4 Marks)

(ii) Using your discretization output in (i), write x_i at each interior nodes, t_i. Hence, transform your system of linear equations into tridiagonal matrix.

(7 Marks)

(iii) Find the approximate solution of the interior position as given by tridiagonal matrix in part (ii) using Thomas algorithm method.

(13 Marks)

[Hint: use radian mode in your calculator.]

[24 Marks, CO2/PO2]

END OF QUESTION PAPER

APPENDIX

Errors

True Error

$$E_t = \text{true value - approximation value}$$

True percent relative error

$$\varepsilon_t = \left|\frac{\text{true value} - \text{approximation value}}{\text{true value}}\right| \times 100$$

Approximate percent relative error

$$\varepsilon_a = \left|\frac{\text{present approximation} - \text{previous approximation}}{\text{present approximation}}\right| \times 100$$

Stopping criterion

Terminate computation when $|\varepsilon_a| < \varepsilon_s$

Roots of Equations

Bisection method

$$x_r = \frac{(x_l + x_u)}{2}$$

False-position method

$$x_r = x_u - \frac{f(x_u)(x_l - x_u)}{f(x_l) - f(x_u)}$$

Secant method

$$x_{i+1} = x_i - \frac{f(x_i)(x_{i-1} - x_i)}{f(x_{i-1}) - f(x_i)}$$

Newton-Raphson method

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

Linear Algebraic Equations and Matrices

System of linear algebraic equations

$[A]\{X\} = \{B\}$. Decomposition $[A] = [L][U]$ with $[L]$ and $[U]$ can be obtained as follows:

Doolittle decomposition

$$[L] = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix}; \ [U] = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix}$$

Cholesky method

$$[A] = [U]^T[U]\ ; \quad [U] = \begin{bmatrix} u_{11} & u_{12} & u_{13} \\ 0 & u_{22} & u_{23} \\ 0 & 0 & u_{33} \end{bmatrix},$$

$$u_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} u_{ki}^2}$$

$$u_{ij} = \frac{a_{ij} - \sum_{k=1}^{i-1} u_{ki}u_{kj}}{u_{ii}} \quad \text{for } j = i+1, \ldots, n$$

Crout's method

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} = \begin{bmatrix} l_{11} & 0 & 0 & 0 \\ l_{21} & l_{22} & 0 & 0 \\ l_{31} & l_{32} & l_{33} & 0 \\ l_{41} & l_{42} & l_{43} & l_{44} \end{bmatrix} \begin{bmatrix} 1 & u_{12} & u_{13} & u_{14} \\ 0 & 1 & u_{23} & u_{24} \\ 0 & 0 & 1 & u_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Jacobi method

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left[b_i - \left(\sum_{\substack{j=1 \\ j\neq i}}^{j=n} a_{ij} x_j^{(k)} \right) \right], i = 1, 2, ..., n.$$

Gauss seidel method

$$x_i^{(k+1)} = \frac{b_i}{a_{ii}} - \sum_{j=1}^{i-1} \frac{a_{ij}}{a_{ii}} x_j^{(k+1)} - \sum_{j=i+1}^{n} \frac{a_{ij}}{a_{ii}} x_j^{(k)}$$

where

$k = 1, 2, \ldots$

$i = 1, 2, \ldots, n$

Power method

$$v^{(k+1)} = \frac{1}{m_{k+1}} A v^{(k)}$$

$k = 0,1,2,\ldots.$

Nonlinear system: Newton-Raphson method

$$x_{1,i+1} = x_{1,i} - \frac{f_{1,i} \frac{\partial f_{2,i}}{\partial x_2} - f_{2,i} \frac{\partial f_{1,i}}{\partial x_2}}{|J|}$$

$$x_{2,i+1} = x_{2,i} - \frac{f_{2,i} \frac{\partial f_{1,i}}{\partial x_1} - f_{1,i} \frac{\partial f_{2,i}}{\partial x_1}}{|J|}$$

$$[J] = \begin{bmatrix} \frac{\partial f_{1,i}}{\partial x_1} & \frac{\partial f_{1,i}}{\partial x_2} \\ \frac{\partial f_{2,i}}{\partial x_1} & \frac{\partial f_{2,i}}{\partial x_2} \end{bmatrix}$$

Curve Fitting

Newton interpolation polynomial

$$f_n(x) = f(x_0) + f[x_0, x_1](x - x_0)$$
$$+ f[x_0, x_1, x_2](x - x_0)(x - x_1)$$
$$+ \cdots + f[x_0, \ldots, x_{n-1}, x_n](x - x_0)(x - x_1)\cdots(x - x_{n-1})$$

where

$$f[x_0, \ldots, x_{n-1}, x_n] = \frac{f[x_n, x_{n-1}, \ldots, x_1] - f[x_{n-1}, x_{n-2}, \ldots, x_0]}{x_n - x_0}$$

Lagrange interpolation polynomial

$$f_n(x) = \sum_{i=0}^{n} L_i(x) f(x_i) \text{ where } L_i(x) = \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j}$$

$n -$ order of interpolation

Inverse Newton interpolation polynomial

$$P_n(f) = x_0 + b_1(f - f_0) + b_2(f - f_0)(f - f_1) + b_3(f - f_0)(f - f_1)(f - f_2) + \cdots + b_n(f - f_0)(f - f_1)\cdots(f - f_{n-1})$$

n – order of interpolation

Inverse Lagrange interpolation polynomial

$$P_n(f) = \sum_{i=0}^{n} L_i(f) x_i \quad \text{where} \quad L_i(f) = \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{(f - f_j)}{(f_i - f_j)}$$

n – order of interpolation

Linear Splines

$$s_i(x) = f(x_i) + \frac{f_{i+1} - f_i}{x_{i+1} - x_i}(x - x_i) \qquad x_i \le x \le x_{i+1}$$

Quadratic Splines

$$s_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 \qquad x_i \le x \le x_{i+1}$$

For $i = 1, 2, \ldots, n-1$, find

$h_i = x_{i+1} - x_i$; $\quad f_i + b_i h_i + c_i h_i^2 = f_{i+1}$

$b_i + 2c_i h_i = b_{i+1}$;

Also given,

$c_1 = 0$

$a_i = f_i$

Numerical Integration

Trapezoidal rule

$$I \cong \frac{h}{2}\left[f(x_0) + 2\sum_{i=1}^{n-1} f(x_i) + f(x_n) \right]$$

where

$$h = \frac{x_n - x_0}{n}$$

Simpson's 1/3rd rule

$$I \cong \frac{h}{3}\left[f(x_0) + 4\sum_{i=1,3,5}^{n-1} f(x_i) + 2\sum_{j=2,4,6}^{n-2} f(x_j) + f(x_n) \right]$$

where

$h = \dfrac{x_n - x_0}{n}$ and n must even segment

Simpson's 3/8 rule

$$I \cong \frac{3h}{8}\left[f(x_0) + 3f(x_1) + 3f(x_2) + f(x_3) \right], \text{ where } h = \frac{x_3 - x_0}{3}$$

Ordinary Differential Equations (IVP)

Euler's method

$$y_{i+1} = y_i + hf(x_i, y_i)$$
$$x_{i+1} = x_i + h$$

2nd order Runge-Kutta: Heun method

$$y_{i+1} = y_i + \frac{1}{2}(k_1 + k_2)h$$
$$x_{i+1} = x_i + h$$
where
$$k_1 = f(x_i, y_i)$$
$$k_2 = f(x_i + h, y_i + k_1 h)$$

2nd order Runge-Kutta: Midpoint method

$$y_{i+1} = y_i + k_2 h$$
$$x_{i+1} = x_i + h$$
where
$$k_1 = f(x_i, y_i)$$
$$k_2 = f(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_1 h)$$

2nd order Runge-Kutta: Ralston's method

$$y_{i+1} = y_i + \frac{1}{3}(k_1 + 2k_2)h$$
$$x_{i+1} = x_i + h$$

where
$$k_1 = f(x_i, y_i)$$
$$k_2 = f(x_i + \frac{3}{4}h, y_i + \frac{3}{4}k_1 h)$$

Fourth order Runge-Kutta method

$$y_{i+1} = y_i + \frac{h}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right)$$
$$x_{i+1} = x_i + h$$
where
$$k_1 = f\left(x_i, y_i\right)$$
$$k_2 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_1 h\right)$$
$$k_3 = f\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_2 h\right)$$
$$k_4 = f\left(x_i + h, y_i + k_3 h\right)$$

Ordinary Differential Equations (BVP)

Shooting method

Extrapolate estimate for initial slope

$$z(0) = G1 + \frac{G2 - G1}{R2 - R1}(D - R1)$$

where

G1 = First guess at initial slope
G2 = Second guess at initial slope
R1 = Final result at endpoint (using G1)
R2 = Second result at endpoint (using G2)
D = the desired value at the endpoint

Finite Difference method

$$\frac{d^2y}{dx^2} = \frac{y_{i+1} - 2y_i + y_{i-1}}{\Delta x^2}$$

$$\frac{dy}{dx} = \frac{y_{i+1} - y_{i-1}}{2\Delta x}$$

$$\left(1 - \frac{\Delta x}{2} p_i\right) y_{i-1} - \left(2 - \Delta x^2 q_i\right) y_i + \left(1 + \frac{\Delta x}{2} p_i\right) y_{i+1} = \Delta x^2 r_i$$

Thomas Algorithm

$$\alpha_i = d_i - c_i \beta_{i-1} \quad , \quad \alpha_1 = d_1$$

$$\beta_i = \frac{e_i}{\alpha_i}$$

$$w_i = \frac{b_i - c_i w_{i-1}}{\alpha_i} \quad , \quad w_1 = \frac{b_1}{\alpha_1}$$

$$y_i = w_i - \beta_i y_{i+1} \quad , \quad y_4 = w_4$$